

Universidade Federal do Rio de Janeiro

Núcleo de Computação Eletrônica

Alex Avellar de Oliveira

METROLOGIA DE REDES:

**Estimativa da Capacidade do Caminho em
uma Rede Embutido no Protocolo TCP**

Rio de Janeiro

2010

ALEX AVELLAR DE OLIVEIRA

**METROLOGIA DE REDES: Estimativa da Capacidade do
Caminho em uma Rede Embutido no Protocolo TCP**

Monografia apresentada para obtenção do título de Especialista em Gerência de Redes de Computadores no Curso de Pós-Graduação Lato Sensu em Gerência de Redes de Computadores e Tecnologia Internet do Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro – NCE/UFRJ.

Orientador:

Moacyr Henrique Cruz de Azevedo, M.Sc., UFRJ, Brasil

Rio de Janeiro

2010

Alex Avellar de Oliveira

**METROLOGIA DE REDES: Estimativa da Capacidade do
Caminho em uma Rede Embutido no Protocolo TCP**

Monografia apresentada para obtenção do título de Especialista em Gerência de Redes de Computadores no Curso de Pós-Graduação Lato Sensu em Gerência de Redes de Computadores e Tecnologia Internet do Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro – NCE/UFRJ.

Aprovada em março de 2010.



Moacyr Henrique Cruz de Azevedo, M.Sc., UFRJ, Brasil

RESUMO

OLIVEIRA, Alex Avellar de. **METROLOGIA DE REDES: Estimativa da capacidade do caminho em uma rede embutido no protocolo TCP**. Monografia (Especialização em Gerência de Redes e Tecnologia Internet). Núcleo de Computação Eletrônica, Universidade Federal do Rio de Janeiro. Rio de Janeiro, 2010.

A Metrologia de Rede consiste em medir a performance na rede e a dificuldade aumenta de acordo com o crescimento dessas como a Internet. Determinadas aplicações na Internet funcionariam melhor se conseguissem obter características dos caminhos por onde passam.

Existe um grande número de pesquisas técnicas que objetivam medir essas características. Como exemplo, pode-se estimar a capacidade do caminho em uma rede, estimar a disponibilidade de largura de banda, identificar melhores caminhos considerando o balanceamento de carga e aferir o desempenho de um caminho no que tange a perda, a retardo unidirecional, a RTT (Round Trip Time) e jitter.

São fatores determinantes na obtenção de características na metrologia das redes: assumir a inexistência de ajuda dos equipamentos das redes que estão no meio do caminho; utilizar o mínimo de tráfego e processamentos adicionais, ou seja, baixo custo operacional; e, principalmente, manter a rapidez na obtenção das características, o suficiente para que uma aplicação possa usufruir dos resultados.

O presente trabalho reproduz e compara as principais soluções encontradas que viabilizam a estimação de capacidade de um caminho utilizando o Transmission Control Protocol (TCP). Utilizou-se a técnica criada pelo grupo de medições da Universidade Federal do Espírito Santo, TCP-ECAP (Embedded Path Capacity Estimator in TCP) usado de forma passiva, sem o uso de pacotes de pesquisa. A estimativa de capacidade é obtida realizando inferências no protocolo TCP respeitando o mínimo de mudanças a partir do método de obtenção da estimativa de capacidade por par de pacotes.

Foram realizadas alterações em arquivos da aplicação TCP no linux resultando na necessidade de compilação do Kernel. A rede de dados do Exército Brasileiro foi utilizada para os experimentos da ferramenta, com a seleção de Web Servers localizados em Brasília, São Paulo, Porto Alegre e Rio de Janeiro. Foram realizados testes com um gerador de gráficos denominado *Pathcrawler* com o objetivo de melhorar visualmente os resultados.

ABSTRACT

OLIVEIRA, Alex Avellar de. **METROLOGIA DE REDES: Estimativa da capacidade do caminho em uma rede embutido no protocolo TCP**. Monografia (Especialização em Gerência de Redes e Tecnologia Internet). Núcleo de Computação Eletrônica, Universidade Federal do Rio de Janeiro. Rio de Janeiro, 2010.

The Metrics Network is to measure the network performance and the difficulty increases with the growth of these as the Internet. Certain applications on the Internet would work best if they could obtain characteristics of trodden paths.

There are a number of research techniques that aim to measure these characteristics. As an example, we can estimate the capacity of the path in a network, estimate the available bandwidth, identify the best ways considering the load balancing and measuring the performance of a path with respect to loss, one-way delay, the RTT (Round Trip Time) and jitter.

Are determining factors in obtaining metrology characteristics of networks, and take no help from the network equipment that are on the way, use the least traffic and further processing, ie, low cost, and above all, keep timeliness of the features, enough for one application can use the results.

This work reproduces and compares the main solutions that enable the estimation of capacity of a path using the Transmission Control Protocol (TCP). We used the technique created by the group of measurements of the Federal University of Espirito Santo, TCP-ECAP (Embedded Path Capacity Estimator in TCP) used passively, without the use of search packages and the estimated capacity is obtained by performing inferences in the protocol TCP respecting the minimum number of changes from the method of obtaining the estimate of capacity per pair of packets.

There have been changes in the application files in Linux TCP resulting in the need to compile a Kernel. The data network of the Brazilian Army was used for the experiments of the tool, with the selection of Web servers located in Brasilia, São Paulo, Porto Alegre and Rio de Janeiro. Tests were conducted with a graphics generator called *Pathcrawler* in order to visually enhance the results.

LISTA DE FIGURAS

	Página
Figura 1 – Capacidade do enlace	15
Figura 2 – Dispersão do par de pacotes	17
Figura 3 – Uso da técnica VPS com a ferramenta <i>Pathchar</i>	18
Figura 4 – Fase <i>slow-start</i> no controle de congestionamento TCP	21
Figura 5 – Algoritmo <i>Short Regulated Rate</i>	24
Figura 6 – Algoritmo <i>No Short Regulated Rate</i>	25
Figura 7 – Enlaces da rede na cidade de do Rio de Janeiro	27
Figura 8 – Infraestrutura própria da rede	28
Figura 9 – Experimentos dentro da rede do Exército	34
Figura 10 – Experimentos à Internet com enlaces de menor capacidade	36
Figura 11 – Estimativa de MinRTT entre localidades	37
Figura 12 – Gráfico de distribuição das capacidades	38
Figura 13 – Gráfico de distribuição MinRTT	38

LISTA DE TABELAS

	Página
Tabela 1 – Rede de testes (São Paulo)	29
Tabela 2 – Rede de testes (Porto Alegre)	29
Tabela 3 – Rede de testes (Brasília)	30
Tabela 4 – Resultados São Paulo/Rio de Janeiro	32
Tabela 5 – Resultados Porto Alegre/Rio de Janeiro	32
Tabela 6 – Resultados Brasília/Rio de Janeiro	33

LISTA DE ABREVIATURAS E SIGLAS

ACK	Acknowledge
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
RTT	Round Trip Time
SLOPs	Self Loading Periodic Streaming
TCP	Transmission Control Protocol
TOPP	Trains of Packets Pairs
VPS	Variable Packet Size Probing

SUMÁRIO

	Página
1 INTRODUÇÃO	10
1.1 PROBLEMATIZAÇÃO	10
1.2 JUSTIFICATIVA DO TEMA	10
1.3 HIPÓTESE/OBJETIVO	11
1.4 RESULTADOS ESPERADOS	11
2 REFERENCIAL TEÓRICO	12
3 METODOLOGIA	14
4 TÉCNICAS DE METROLOGIA DE REDES	15
4.1 ESTIMATIVA DA CAPACIDADE DO ENLACE	15
4.2 MEDIÇÕES ATIVAS X PASSIVAS	16
4.3 A TÉCNICA DA ESTIMATIVA USANDO PAR DE PACOTES	16
4.4 VPS	18
4.5 SLOPs	19
4.6 TOPP	19
4.7 PARIS TRACEROUTE	19
5 EMBUTINDO A ESTIMATIVA DE CAPACIDADE NO PROTOCOLO TCP	20
5.1 A APLICAÇÃO TCP-ECAP	20
5.2 ALGORITMO DE INFERÊNCIA	21
5.3 ALGORITMO SHORT REGULATED RATE	23
5.4 ALGORITMO NO SHORT REGULATED RATE	24
6 EXPERIMENTOS COM A TÉCNICA DO PAR DE PACOTES	26
6.1 INSTRUMENTAÇÃO DO KERNEL	26
6.2 CENÁRIO CRIADO PARA OS EXPERIMENTOS	26
6.3 PORTABILIDADE E FERRAMENTAS UTILIZADAS NOS TESTES	30
6.3.1 Portabilidade	30
6.3.2 Ferramentas para captura e análise	31
7 RESULTADOS DOS EXPERIMENTOS COM A TÉCNICA DO PAR DE PACOTES USANDO O TCP-ECAP	32
8 RESULTADOS DOS EXPERIMENTOS COM O USO DO GERADOR DE GRÁFICOS – PATHCRAWLER	34
9 CONCLUSÃO	39
10 TRABALHOS FUTUROS	41
11 REFERÊNCIAS	42

1 INTRODUÇÃO

1.1 PROBLEMATIZAÇÃO

Determinadas aplicações na Internet funcionariam melhor se conseguissem obter características dos caminhos por onde passam. Existe um grande número de pesquisas técnicas que objetivam medir essas características. Como exemplo, pode-se estimar a capacidade do caminho em uma rede, estimar disponibilidade e largura de banda, identificar melhores caminhos considerando o balanceamento de carga, e aferir o desempenho de um caminho no que tange a perda, retardo unidirecional, RTT (Round Trip Time) e jitter.

São fatores determinantes na obtenção de características na metrologia das redes: assumir a inexistência de ajuda dos equipamentos das redes que estão no meio do caminho; utilizar o mínimo de tráfego e processamentos adicionais, ou seja, baixo custo operacional; e, principalmente, manter a rapidez na obtenção das características, o suficiente para que uma aplicação possa usufruir os resultados.

1.2 JUSTIFICATIVA DO TEMA

A estimativa da capacidade de um caminho na Internet é fundamental para diversas aplicações como Multimídia Streaming, Planejamento de Redes, Redes Sobrepostas (Overlay Networks) e Controle de Congestionamento, pois essas aplicações desejam otimizar recursos. O presente trabalho reproduz e compara as principais soluções encontradas que viabilizam a estimação de capacidade de um caminho utilizando o Transmission Control Protocol sem a necessidade de acesso ou permissão em elementos ativos remotos no caminho da rede. Utilizou-se o método TCP-ECAP (Embedded Path Capacity Estimator in TCP) usado de forma

passiva, sem o uso de pacotes de pesquisa e a estimativa de capacidade é obtida realizando inferências no protocolo TCP respeitando o mínimo de mudanças.

Foram realizadas alterações em arquivos da aplicação TCP no linux resultando na necessidade de compilação do Kernel. A rede de dados do Exército Brasileiro foi utilizada para testes da ferramenta, com a seleção de Web Servers localizados em Brasília, São Paulo, Porto Alegre e Rio de Janeiro e o mapeamento da capacidade de banda dos caminhos. Para fornecer uma capacidade maior a ferramenta, a fim de aplicá-la em diferentes ambientes independente da plataforma de hardware foi criado um sistema com portabilidade.

1.3 HIPÓTESE/OBJETIVO

Os propósitos do trabalho são, primeiramente, realizar o cruzamento das técnicas de metrologia com o cenário escolhido, focando na análise do desempenho destas técnicas mediante os cenários, a fim de produzir resultados e compará-los com as características reais da rede, e avaliar uma nova ferramenta que usa os resultados estimados pelo método TCP-ECAP e forma gráficos indicando a capacidade dos enlaces, denominada *PathCrawler*.

1.4 RESULTADOS ESPERADOS

Descrição das técnicas existentes, com a análise de desempenho e a comparação da eficiência da técnica do par de pacotes através da ferramenta TCP-ECAP no cenário escolhido, levantando os problemas e deficiências ainda não somados.

Propor soluções de problemas através da análise de experimentos no cenário da rede do Exército e da Internet com a ferramenta de criação de gráfico de capacidade de banda PathCrawler, correlacionando métricas e avaliando seu desempenho.

2 REFERENCIAL TEÓRICO

Trabalhos anteriores sobre estimativa de capacidade de um caminho foram baseados no uso da dispersão de pares de pacotes, ou seja, o intervalo entre os pacotes após deixarem o enlace que apresenta gargalo. [Carter 1996] propôs o *Bprobe*, onde métodos de filtragem são aplicados na medição do par de pacotes de diferentes tamanhos. [Lai e Baker 1999] usaram a medição por par de pacotes, mas os filtraram através de um algoritmo de densidade do Kernel [Scott 1992], estatisticamente válido, relativamente simples e de rápida computação.

Outras técnicas que não foram baseadas na dispersão de pares de pacotes, mas na variação do atraso round-trip com o aumento do tamanho do pacote, foram usadas por [Jacobson 1997] na ferramenta *Pathchar* que se baseia na geração de respostas ICMP de roteadores, onde tenta estimar a capacidade de um enlace enviando um conjunto de pacotes não pares no enlace, com diferentes tamanhos. Os atrasos mínimos para diferentes tamanhos de pacotes formam uma equação linear que produz a largura de banda de um enlace. *Pathchar* também é conhecida por consumir uma quantidade significativa da quantidade de largura de banda [Lai e Baker 1999].

As técnicas citadas somente conseguem descobrir atrasos ou apenas a dispersão dos pacotes de pesquisa e não os dois juntos. *CapProbe*, [Kappor 2004], combina a dispersão e medição de atrasos utilizando pares de pacotes calculando a capacidade do caminho e a localização do pacote. *CapProbe* requer acesso ao outro lado da rede, ou seja ao transmissor, um fato que limita consideravelmente a operação. O *Pathrate* pode ser executado em ambos os lados, porém trabalha na camada de aplicação, perdendo assim em precisão das medidas, enquanto o

CapProbe é executado no modo Kernel, trabalhando assim com altas prioridades e precisão. Outro problema encontrado no *Pathrate* é o tempo, pois depende de uma grande análise estatística, enquanto *CapProbe* necessita de apenas alguns segundos.

3 METODOLOGIA

A fase inicial deste trabalho envolve a identificação dos métodos existentes e do cenário de uso para as diversas soluções em Metrologia de redes.

A segunda fase é a escolha da técnica do par de pacotes através da ferramenta TCP-ECAP para ser aplicada no cenário de aplicação escolhido.

A terceira fase é a análise de experimentos realizados no âmbito da Rede do Exército, com implantação das últimas ferramentas existentes de cada técnica classificada, aproveitando o fato de ser realizado em um ambiente controlado.

A quarta fase é a análise de novos experimentos usando um conjunto de ferramentas para apresentação de gráficos contendo a capacidade de banda nos enlaces pesquisados em um formato de melhor visualização, e novos testes usando a Internet a fim de avaliar as técnicas em um ambiente mais amplo.

O levantamento de resultados e deficiências será a quinta fase deste estudo, identificando os problemas dentro dos cenários escolhidos a fim de produzir recomendações de uso para cada cenário.

4 TÉCNICAS DE METROLOGIA DE REDES

É possível enumerar técnicas de medições e suas ferramentas apresentadas por trabalhos anteriores. Para algumas técnicas existem ferramentas que trazem vantagens e algumas limitações, porém são importantes para o melhor entendimento das medições a fim de melhor definir a metrologia de redes.

4.1 ESTIMATIVA DA CAPACIDADE DO ENLACE

Várias pesquisas estão sendo desenvolvidas para a área de estimativa de capacidade. A capacidade de um enlace pode ser definida como a taxa máxima que pode ser alcançada em uma conexão na ausência de qualquer tráfego, ou seja, qual a quantidade máxima em bps que pode ser enviado por determinado enlace. É importante definir que a capacidade do enlace é diferente da largura de banda disponível. Enquanto na capacidade do enlace temos a taxa máxima de transmissão de bits do enlace, na largura de banda temos a taxa de transferência disponível para uma aplicação em determinado instante, em outras palavras a porção da capacidade de banda que não está sendo usada por tráfego concorrente.

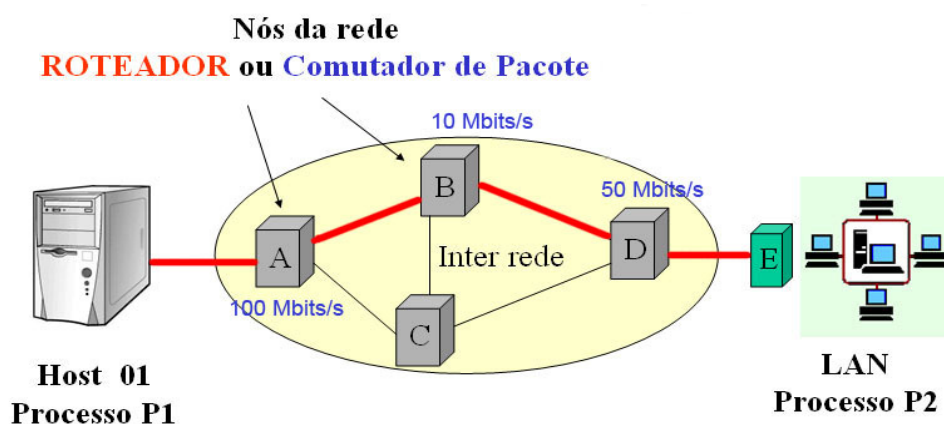


Figura 1 – Capacidade do enlace

O enlace de menor capacidade no caminho determina a capacidade fim-a-fim de todo o enlace. A figura 1 mostra um caminho fim-a-fim com capacidade limitada pelo enlace de menor capacidade, ou seja 10Mbps/s.

4.2 MEDIÇÕES ATIVAS X PASSIVAS

Uma característica importante das técnicas de estimativa conhecidas é a forma como elas trabalham. Podem ser identificadas como ativas e passivas. As ativas são as que geram algum tipo de tráfego extra para conseguir resultados, porém trazem a vantagem de serem mais flexíveis para projetar pacotes de pesquisa, com propriedades particulares a fim de igualar as requisições de medições tais como média de atraso, perdas no roteador, capacidade e largura de banda disponível. As medições passivas não geram pacotes extras para obter suas medições, não aumentam o tráfego e em alguns casos não necessitam de administração em qualquer lado do caminho que se deseja estimar. Porém, é pouco flexível quando é necessário projetar pacotes com informações mais específicas, sendo mais indicada em redes onde não é possível a administração dos equipamentos envolvidos.

4.3 A TÉCNICA DA ESTIMATIVA USANDO PAR DE PACOTES

A PPTD (Packet Pair/Train Dispersion Probing) conhecida como dispersão dos pares de pacotes, ou seja o intervalo entre pares de pacotes, é uma técnica utilizada por trabalhos, que medem a capacidade do menor enlace ou a largura de banda. A figura 2 representa este método.

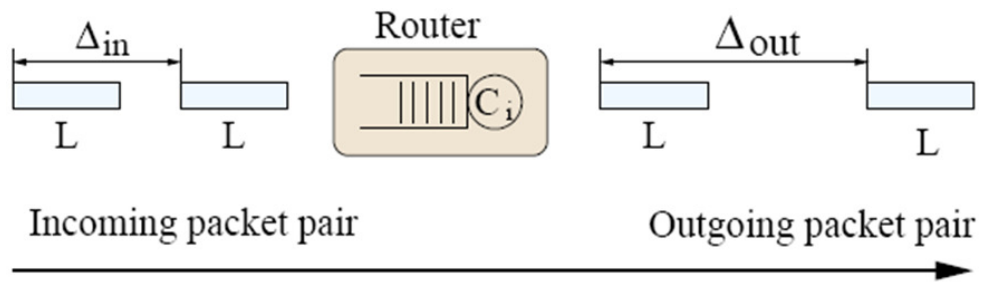


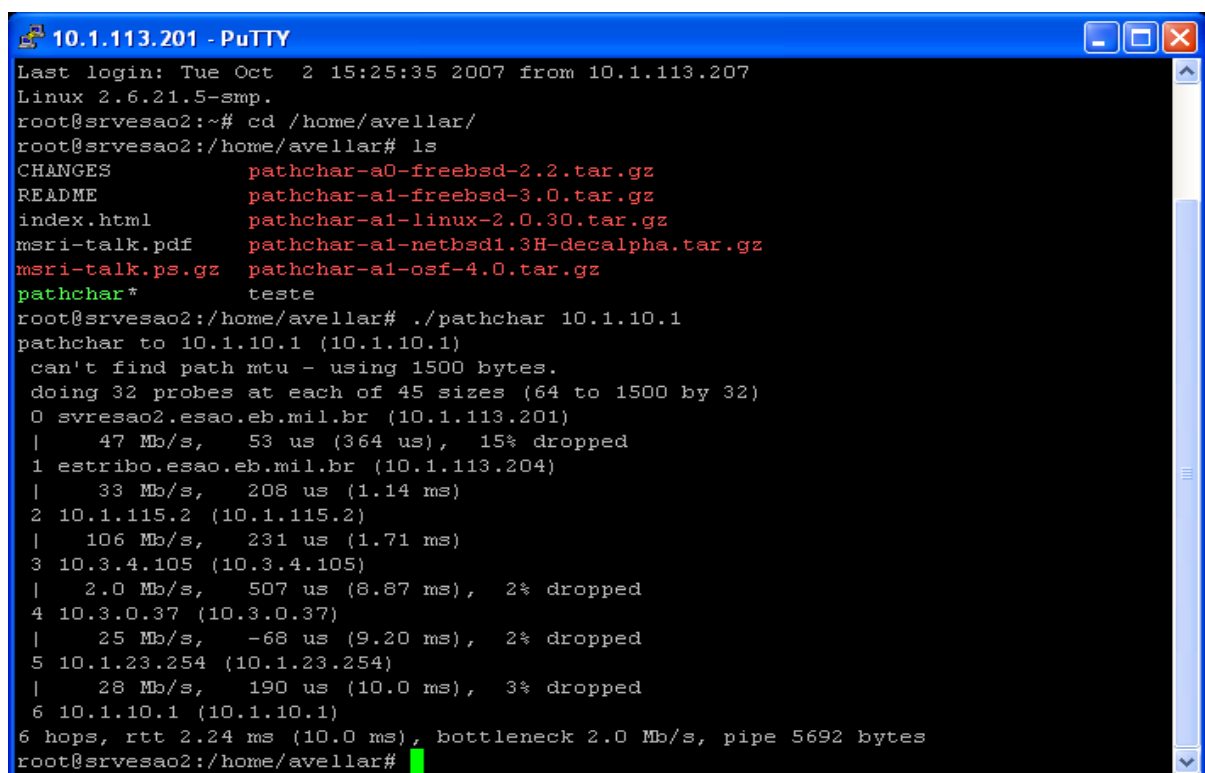
Figura 2 – Dispersão do par de pacotes

[Carter 1996] propôs a ferramenta *Bprobe*, onde métodos de filtragem são aplicados na medição do par de pacotes de diferentes tamanhos. [Lai e Baker 1999] usaram a medição por par de pacotes, mas os filtraram através de um algoritmo de densidade do Kernel [Scott 1992], estatisticamente válido, relativamente simples e de rápida computação. Uma ferramenta denominada *CapProbe*, combina a possibilidade de obter a estimativa de capacidade pela dispersão dos pacotes com a indicação do local do gargalo. O problema do *CapProbe* está no fato de necessitar de acesso ao outro lado da rede, ou seja o lado transmissor, um fato que limita consideravelmente a operação.

Uma ferramenta apresentada por [Dovrolis-Prasad 2001] denominada *Pathrate*, pode ser executada em ambos os lados, porém trabalha na camada de aplicação, diminuindo assim a precisão das medidas uma vez que o *CapProbe* é executado em modo Kernel. Outro ponto contra do *Pathrate* é o tempo, uma vez que depende de uma grande análise estatística. Uma das ferramentas mais recentes que usa a técnica dos pares de pacote é chamada de TCP-ECAP [Martinello 2007] diferente das outras ferramentas, esta aproveita pacotes enviados pelo protocolo TCP sem o uso de pacotes extras em suas medições.

4.4 VPS

Outra técnica conhecida é classificada como VPS (Variable Packet Size Probing). Como exemplo de ferramentas que usam essa técnica temos a *Pathchar* [Jacobson 1997] que é uma técnica que usa pacotes de tamanhos variáveis para encontrar a capacidade para cada nó, no caso o *Pathchar* usa a TTL dos pacotes ICMP que expira em determinados momentos.



```

10.1.113.201 - PuTTY
Last login: Tue Oct  2 15:25:35 2007 from 10.1.113.207
Linux 2.6.21.5-smp.
root@srvesao2:~# cd /home/avellar/
root@srvesao2:/home/avellar# ls
CHANGES          pathchar-a0-freebsd-2.2.tar.gz
README            pathchar-a1-freebsd-3.0.tar.gz
index.html        pathchar-a1-linux-2.0.30.tar.gz
msri-talk.pdf     pathchar-a1-netbsd1.3H-decalpha.tar.gz
msri-talk.ps.gz   pathchar-a1-osf-4.0.tar.gz
pathchar*         teste
root@srvesao2:/home/avellar# ./pathchar 10.1.10.1
pathchar to 10.1.10.1 (10.1.10.1)
can't find path mtu - using 1500 bytes.
doing 32 probes at each of 45 sizes (64 to 1500 by 32)
0 svresao2.esao.eb.mil.br (10.1.113.201)
| 47 Mb/s, 53 us (364 us), 15% dropped
1 estribo.esao.eb.mil.br (10.1.113.204)
| 33 Mb/s, 208 us (1.14 ms)
2 10.1.115.2 (10.1.115.2)
| 106 Mb/s, 231 us (1.71 ms)
3 10.3.4.105 (10.3.4.105)
| 2.0 Mb/s, 507 us (8.87 ms), 2% dropped
4 10.3.0.37 (10.3.0.37)
| 25 Mb/s, -68 us (9.20 ms), 2% dropped
5 10.1.23.254 (10.1.23.254)
| 28 Mb/s, 190 us (10.0 ms), 3% dropped
6 10.1.10.1 (10.1.10.1)
6 hops, rtt 2.24 ms (10.0 ms), bottleneck 2.0 Mb/s, pipe 5692 bytes
root@srvesao2:/home/avellar#

```

Figura 3 – Uso da técnica VPS com a ferramenta *Pathchar*

A figura 3 indica um resultado após a utilização desta ferramenta em uma rede de teste, verificando-se que neste caso temos a capacidade do gargalo que na figura aparece como de 2Mb/s.

4.5 SLOPs

A técnica denominada SLOPs (Self Loading Periodic Streaming), envia pacotes de tamanhos iguais em uma taxa constante e mede a disponibilidade de largura de banda no caminho fim-a-fim. Ferramentas como *Pathload* e IGI usam essa técnica. A ferramenta mais recente que trabalha com esse método é chamada *Spruce*, um trabalho de [Jacob 2004] que apresenta essa ferramenta e compara com as outras duas citadas e usa o MRTG para validar os testes. Os resultados das pesquisas mostram que as ferramentas que trabalham com a técnica SLOPs não apresentam resultados precisos quando usados em enlaces de alta velocidade pois os resultados mostram elevadas discrepâncias.

4.6 TOPP

A técnica TOPP (Trains of Packets Pairs), estima a disponibilidade de largura de banda. Para isso envia muitos pares de pacotes em taxas crescentes da origem para um destino.

4.7 PARIS TRACEROUTE

Ferramenta que propõem uma metodologia para realizar diagnósticos de rede, trabalhar com medições e identificar problemas causados por balanceamento de cargas. A ferramenta usa o cabeçalho de pacotes de pesquisa para controlar e fazer com que esses pacotes sigam o mesmo caminho na presença de um balanceamento de carga, permitindo que o usuário distinga entre um balanceamento de carga por pacotes ou por fluxo.

5 EMBUTINDO A ESTIMATIVA DE CAPACIDADE NO PROTOCOLO TCP

O protocolo TCP tem se mostrado escalável, adaptável e apropriado a uma grande variedade de aplicações [Tanenbaum 1996]. As aplicações existentes não demonstram confiabilidade na medição da capacidade do link. O primeiro objetivo deste trabalho é mostrar que a estimativa de capacidade pode ser obtida realizando inferências no comportamento tradicional do TCP. Para isso foi criado o TCP-ECAP (Embedded Path Capacity Estimator in TCP), que mede no retorno do pacotes a dispersão entre eles durante a fase de *slow-start*. Como os pacotes são enviado de volta (packet pairs), a dispersão é proporcional ao atraso mínimo de transmissão ao longo do caminho. O par de pacotes pode representar o gargalo na capacidade do caminho do enlace.

5.1 A APLICAÇÃO TCP-ECAP

A aplicação TCP-ECAP utiliza dois algoritmos *Short Regulated* e *No Regulated Rate*. Ambos atuam no lado do receptor do TCP e estimam as capacidades dos caminhos no início da conexão. São implementados dentro do código do Linux no módulo TCP com o objetivo de alcançar resultados precisos nas medições. Não existe a necessidade de possuir acesso ou permissão a máquinas remotas.

O uso do TCP-ECAP permite que o receptor limite a quantidade de tráfego na rede, regulando o tamanho da janela e reduzindo o uso do espaço em buffer na presença de um gargalo. TCP-ECAP trabalha em modo passivo, capturando informações de pacotes sem afetar o fluxo da rede.

Outro aspecto importante é que o protocolo TCP foi projetado para ser controlado pelo lado emissor. TCP-ECAP é baseado no lado receptor do TCP, calculando a capacidade com uma precisão de 64 bits usando o registrador RDTSc

presente em diversos processadores atuais. O tempo de resposta é rápido, podendo ser utilizado de forma rápida pelas aplicações.

5.2 ALGORITMO DE INFERÊNCIA

O TCP-ECAP utiliza dois algoritmos de inferência para medir a capacidade dos caminhos da Internet no modo passivo, baseando-se na dispersão do par de pacotes e no método de filtragem *minimum delay sum*, focando na estimativa e em técnicas de filtragem. Um usuário realizando um download de uma página em um servidor WEB pode ser informado sobre a capacidade do presente caminho. A implementação básica do TCP faz com que a conexão seja controlada pelo lado emissor, mas o TCP-ECAP utiliza as medidas em termos de escalabilidade e a conexão passa a ser controlada pelo lado receptor, informando ao emissor o tamanho que deve ser usado na janela.

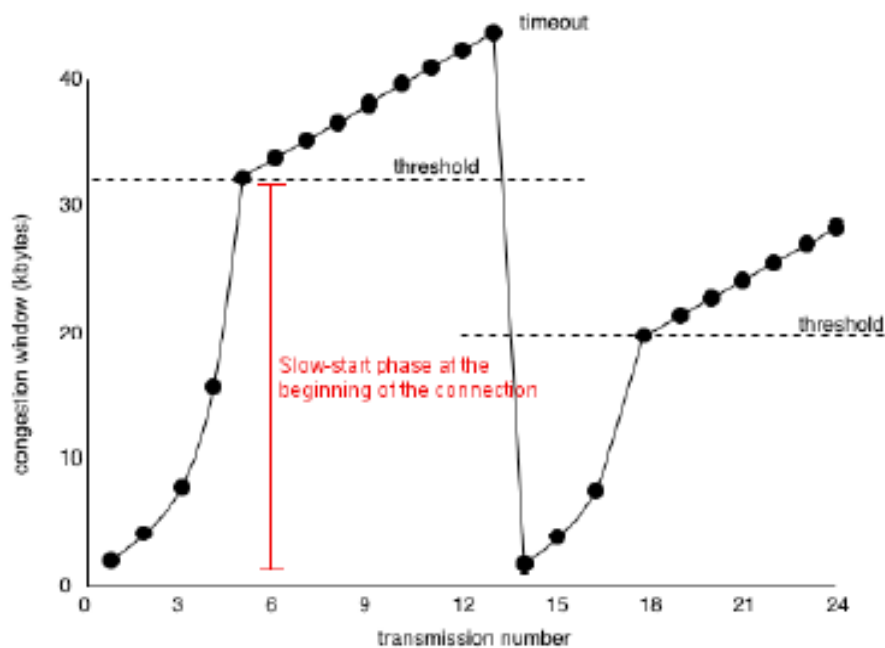


Figura 4 – Fase *slow-start* no controle de congestionamento TCP

Para medir a capacidade é necessário obter pares de pacotes enviado do transmissor para o receptor durante uma conexão TCP normal, e um instante

disponível para obter essas amostras está no início da fase, ou seja, na *slow-start* conhecida como início-lento, descrita na figura 4.

Para alcançar o objetivo, são obtidas amostras de pares de pacotes durante esse comportamento normal na rede. O TCP-ECAP usa o procedimento normal do TCP: para todo ACK enviado do lado receptor do TCP, um par de pacotes é naturalmente transmitido “*back to back*” ao lado transmissor, não mudando o protocolo TCP e não incluindo qualquer pacote extra.

No TCP “tradicional”, quando uma conexão TCP é estabilizada, o emissor inicializa a janela de congestionamento para o tamanho máximo do segmento em uso na conexão e então envia um pacote. Se um ACK é recebido antes do time-out, o TCP adiciona bytes e a janela de congestionamento é aumentada, sendo que para cada ACK recebido o crescimento é exponencial. Na ausência de um ACK, que pode caracterizar um atraso na rede, a janela de congestionamento é reiniciada, retornando a fase de *slow-start*.

O objetivo do TCP-ECAP é estimar a capacidade de caminhos na rede no início da conexão TCP e, assim, softwares e usuários podem usar essas informações rapidamente para seus propósitos no lado cliente. Para realizar esse procedimento o TCP-ECAP, na fase *slow-start*, se aproveita do comportamento normal do TCP

O transmissor TCP prepara dois pacotes para serem transferidos, um correspondendo ao ACK recebido e o outro em referência ao incremento da janela de congestionamento. Logo, para todo ACK enviado do lado do receptor, um par de pacotes é transmitido “*back to back*” no lado transmissor. TCP-ECAP usa esse comportamento normal e executa no lado receptor, identificando tais pares de

pacotes, sendo que a arquitetura do TCP não foi mudada e não foram incluídos quaisquer pacotes de pesquisa.

O *minimum delay sum* (método de filtragem) descarta pacotes não procurados, ou seja, pares que não sofreram atrasos ou enfileiramentos ao longo do caminho. Para cada ciclo RTT o número de pacotes permitidos pela janela de recebimento é dobrado, fazendo assim com que fique difícil o não encontro de pacotes.

São utilizados dois algoritmos o *Short Regulated Rate* e o *No Regulated Rate*.

5.3 ALGORITMO SHORT REGULATED RATE

Na fase *slow-start*, no início de uma conexão TCP, todos os pacotes de dados enviados pelo emissor são respondidos com um pacote ACK pelo lado receptor. O protocolo TCP é projetado para elevar o tamanho da janela do transmissor pelo tamanho de segmentos máximos para cada ACK recebido. Neste caso a janela do receptor é incrementada exponencialmente, dificultando a obtenção de um número suficiente de pares de pacotes para estimar a capacidade.

O algoritmo *Short Regulated Rate* resolve esse problema respondendo com um ACK acumulado para cada x pacotes de dados recebidos, onde $x \geq 2$, ao invés de enviar x ACKs. Quando o emissor recebe um ACK acumulado de x pacotes, ele prepara x pacotes para serem enviados correspondentes aos pacotes de dados recebidos e um único pacote correspondente ao incremento do tamanho da janela, totalizando $x + 1$ pacotes. Diante disso o tamanho da janela do emissor é incrementada para cada x pacotes enviados. O algoritmo então induz uma regulamentação do tamanho da janela do emissor durante a fase de *slow-start*, fornecendo um período de amostras para a estimativa da capacidade conforme mostra a figura 5.

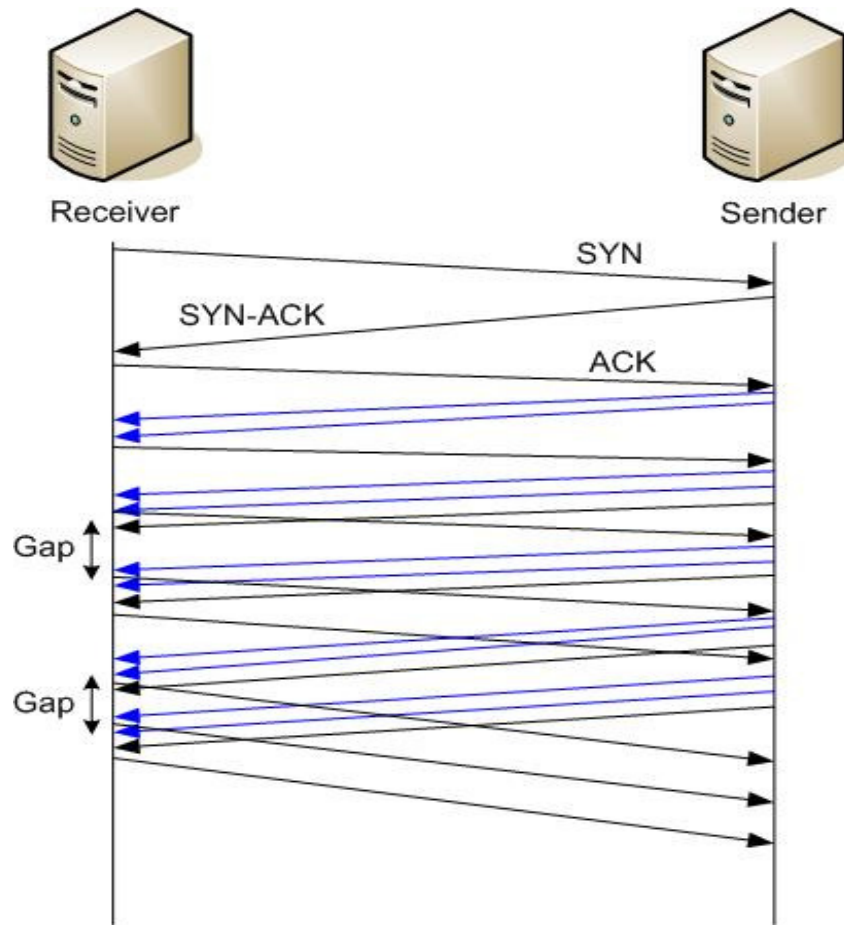


Figura 5 – Algoritmo *Short Regulated Rate*

Esta fase é projetada para coleta de amostras de RTT mínimo em pouco tempo. Baseia-se no fato de que 20 amostras são suficientes para uma correta estimativa de capacidade. Por exemplo, se o RTT for de 50ms, o tempo levado para a coleta de 20 amostras será de 1s, com isso o atraso induzido não compromete a conexão TCP. Após essa fase, o TCP continua incrementando a janela exponencialmente.

5.4 ALGORITMO NO SHORT REGULATED RATE

O objetivo é explorar o comportamento natural da fase *slow-start* do TCP, observando amostras de pares de pacotes que não sofreram atrasos nas filas ao longo do caminho. Ao contrário do algoritmo *Short Regulated Rate*, não existe

nenhum período de regulação imposta e o TCP trabalha normalmente durante toda a conexão, a figura 6 mostra o funcionamento.

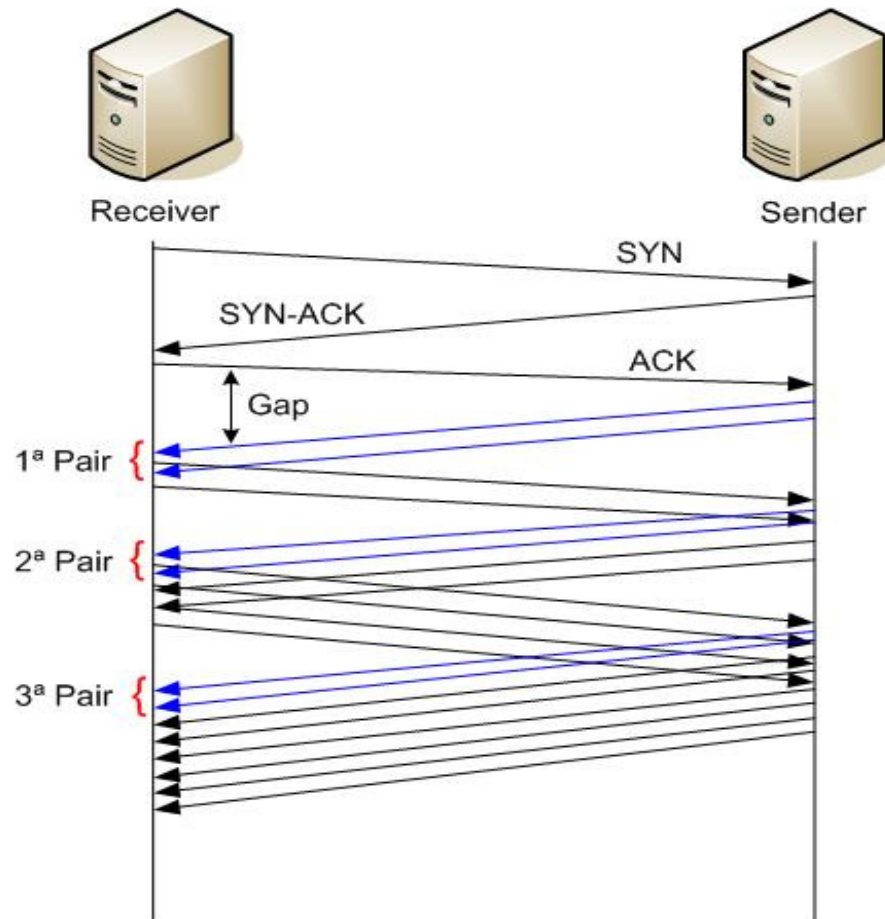


Figura 6 – Algoritmo *No Short Regulated Rate*

Para encontrar pacotes não afetados, é essencial que intervalos entre os ciclos RTT sejam muito bem examinados. A idéia básica é extrair pares no início de cada ciclo RTT do TCP, exatamente o primeiro e o segundo pacote de cada fase.

6 EXPERIMENTOS COM A TÉCNICA DO PAR DE PACOTES

6.1 INSTRUMENTAÇÃO DO KERNEL

A reprodução foi realizada com o *patch* fornecido pelo grupo de medições na UFES [Martinello 2007] denominado *tcp_ecap.dir*, aplicado no Kernel do Linux 2.6.18 onde foram adicionados mudanças para estimar a capacidade. O objetivo é executar TCP-ECAP no modo Kernel, buscando alcançar resultados de alta precisão nas medidas. Os principais arquivos do código TCP estão localizados no diretório *net/ipv4*, exceto arquivos de cabeçalho que estão no diretório *include/net*. São encontrados cinco arquivos nesses locais: *tcp_input.c*, *tcp_output.c*, *tcp_ipv4.c*, *tcp_timer.c* e *tcp.h*. As alterações foram realizadas no *tcp_input.c*, que é responsável por todos os pacotes de dados vindos do transmissor, no *tcp_output.c*, que cuida do envio de ACKs e no arquivo *tcp.h*, que guarda a estrutura de dados compartilhada pelas funções modificadas no TCP.

A instalação inicial foi realizada em uma máquina Dual Intel 3.2Ghz, 1GB RAM, PCI 133Mhz, Intel 1000 Server Pro NIC (1Gbps Ethernet), conectada diretamente a um Cisco Catalyst 3300 com Interface de 1Gbps ligada a uma rede de 100Mbps. Para os testes foram utilizados Web-Servers da rede EBNet, rede de dados do Exército Brasileiro em nível nacional. O Linux Kernel está sendo executado em modo simples para a máquina tipo Desktop e modo super-usuário no notebook.

6.2 CENÁRIO CRIADO PARA OS EXPERIMENTOS

As pesquisas das diferentes técnicas e métodos de metrologia de redes descritos nos principais eventos de fomento desse assunto, até o presente momento, fornecem a possibilidade de classificação desses estudos quanto aos objetivos das medições (capacidade, disponibilidade, identificação e desempenho),

custo operacional (pacotes extras e administração de equipamentos envolvidos) e formas de utilização (aplicações).

A escolha de cenários de uso permite a validação dos métodos pesquisados e a comparação entre as técnicas. Os cenários nos quais a metrologia de rede pode ser aplicada abrangem: as aplicações distribuídas que tomam decisões de roteamento (P2P e Grids); o gerenciamento de redes sobrepostas; a engenharia de tráfego em cima de aplicações sensíveis ao desempenho, como a vídeo-conferência; e o dimensionamento de redes baseado em aplicações.

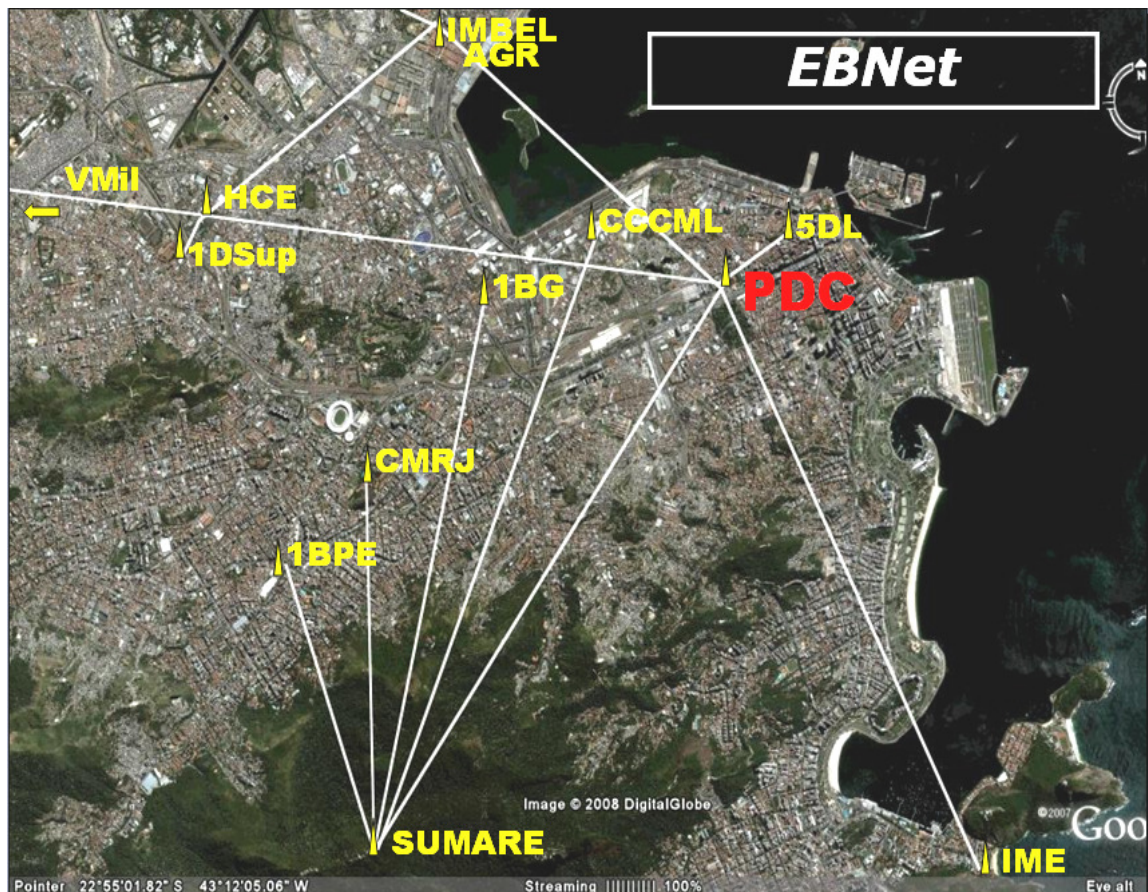


Figura 7 – Enlaces da rede na cidade de do Rio de Janeiro

Como ambiente de testes será utilizada a rede de dados do Exército Brasileiro, com um ambiente heterogêneo de dados, voz e vídeo, foram

selecionados Web-Servers localizados nos seguintes estados: São Paulo (SPO), Porto alegre (POA), Brasília (BSA) e Rio de Janeiro (RJO). A figura 7 mostra alguns enlaces escolhidos para testes dentro da cidade do Rio de Janeiro.

A rede do Exército Brasileiro foi escolhida em virtude de possuir uma quantidade bem significativa de elementos heterogêneos, como enlaces de fibra ótica e uma grande cobertura com enlaces de rádio, diferenciando também na capacidade dos seus enlaces, além da administração de todos os equipamentos e conhecimento da capacidade de todos os enlaces.

A figura 8 mostra alguns exemplos de instalações próprias e equipamentos com possibilidade de administração completa.



Figura 8 – Infraestrutura própria da rede

Os testes foram iniciados do Rio de Janeiro, as tabelas 1, 2 e 3 mostram todos o endereços IP, equipamentos e largura de banda de cada localidade, com destino a máquina com TCP-ECAP compilado que está localizada na cidade do Rio de Janeiro.

Tabela 1 – Rede de testes (São Paulo)

SITE PRINCIPAL (RJO)

Máquina: 10.1.15.47

Link de Saída: 16Mb

SPO - RJO

WebServer: 10.12.4.3

Tamanho do
arquivo: 33,5MB

Link de Saída: 4Mb

Caminho	Equipamento
10.12.4.3	Webserver - SPO
10.22.2.2	Sw L3 Giga
200.182.80.150	Roteador - Rede Embratel MPLS (SPO) - 2Mb
201.45.200.73	Roteador - Rede Embratel MPLS (SPO) - 2Mb
201.45.200.73	Roteador - Rede Embratel MPLS (RJO) - 2Mb
10.2.207.254	Roteador Fast
10.1.15.3	Roteador Giga
10.1.15.47	Máquina com Kernel de módulo TCP-ECAP

Tabela 2 – Rede de testes (Porto Alegre)

POA – RJO

WebServer: 10.25.108.6

Tamanho do arquivo: 3,62MB

Link de Saída: 2Mb

Caminho	Equipamento
10.25.108.6	Webserver - POA
10.25.110.254	Sw L3 Giga
200.248.180.38	Roteador - Rede Embratel MPLS (POA) - 2Mb
200.248.180.37	Roteador - Rede Embratel MPLS (POA) - 2Mb
201.45.200.73	Roteador - Rede Embratel MPLS (RJO) - 2Mb
10.2.207.254	Roteador Fast
10.1.15.3	Roteador Giga
10.1.15.47	Máquina com Kernel de módulo TCP-ECAP

Tabela 3 – Rede de testes (Brasília)

BSA - RJO

WebServer: 10.67.0.4

Tamanho do arquivo: 7,31MB

Link de Saída: 16Mb

Caminho	Equipamento
10.67.0.4	Webserver - BSA
10.77.100.6	Sw L3 Giga
200.252.253.106	Roteador - Rede Embratel MPLS (BSA) - 16Mb
200.252.253.105	Roteador - Rede Embratel MPLS (BSA) - 16Mb
201.45.200.73	Roteador - Rede Embratel MPLS (RJO) - 2Mb
10.2.207.254	Roteador Fast
10.1.15.3	Roteador Giga
10.1.15.47	Máquina com Kernel de módulo TCP-ECAP

6.3 PORTABILIDADE E FERRAMENTAS UTILIZADAS NOS TESTES

6.3.1 Portabilidade

Com o intuito de melhor aproveitar a ferramenta em ambientes diversos de plataforma, o Kernel 2.6.18.3 com TCP-ECAP foi primeiramente instalado em um notebook de menor porte que o Desktop. Após a verificação da boa estabilidade e funcionalidade da ferramenta nessa plataforma foi criado uma ISO com o Kernel 2.6.18.3 e o TCP-ECAP instalado, acompanhado das ferramentas *Wireshark*, TCPTRACE e XPLOT. Um novo ambiente foi criado a partir do Kernel original, as alterações nos arquivos TCP foram feitas de acordo com as necessidades para criação do TCP-ECAP, o novo Kernel foi criado após a compilação, e os pacotes necessários para a instalação das ferramentas citadas foram instalados, gerando assim a imagem e a ISO.

Inicialmente esse trabalho foi tentado na distribuição *Slackware*, porém depois de algumas tentativas frustradas, principalmente na recompilação do Kernel, passou-se a utilizar a distribuição linux *Kurumin* que se mostrou escalável e estável após as modificações. O disco de boot foi testado em máquinas desktops e máquinas virtuais, não apresentou problemas em teste básicos utilizando a ferramenta TCP-ECAP.

6.3.2 Ferramentas para captura e análise

Todos os testes foram realizados com a captura de dados de forma paralela, utilizando o Wireshark para captura de dados com filtragem a pacotes exclusivos do protocolo TCP. Um arquivo foi gerado e comparado de duas formas, no Kernel com módulo TCP-ECAP e no Kernel com módulo TCP normal. Para análise e elaboração de relatórios e gráficos, foi usada a ferramenta TCPTRACE. Os gráficos criados com extensão XPL forma visualizado pela ferramenta XPLOT.

7 RESULTADOS DOS EXPERIMENTOS COM A TÉCNICA DO PAR DE PACOTES USANDO O TCP-ECAP

O grupo de medições da Universidade Federal do Espírito Santo disponibilizou o TCP-ECAP e, de acordo com os resultados obtidos nos testes, observamos uma melhora no download de arquivos quando utilizamos o TCP modificado em relação ao comportamento do TCP original. As tabelas 4, 5 e 6 mostram os valores obtidos na transferência de arquivos em quatro localidades diferentes. Os teste foram realizados durante o funcionamento normal da rede, observando ainda o período de maior pico, entre 14:00 horas e 18:00 horas.

Tabela 4 – Resultados São Paulo/Rio de Janeiro

SPO - RJO

Tamanho do arquivo: 33,5MB

	TCP	TCP-ECAP
Início	3.879.914	0
Término	302.786.524	260.833.464
Tempo Total	298.906.610	260.833.464
	Diferença	Percentual de tempo em relação ao TCP
	-38.073.146	87,26252792

Valores expressos em segundos

Tabela 5 – Resultados Porto Alegre/Rio de Janeiro

POA - RJO

Tamanho do arquivo: 3,62MB

	TCP	TCP-ECAP
Início	3.387.867	4.498.702
Término	33.258.182	31.627.349
Tempo Total	29.870.315	27.128.647
	Diferença	Percentual de tempo em relação ao TCP
	-2.741.668	90,82142924

Valores expressos em segundos

Tabela 6 – Resultados Brasília/Rio de Janeiro

BSA - RJO

Tamanho do arquivo: 7,31MB

	TCP	TCP-ECAP
Início	2.378.055	4.812.214
Término	24.876.392	23.882.807
Tempo Total	22.498.337	19.070.593
	Diferença	Percentual de tempo em relacao ao TCP
	-3.427.744	84,76445615

Valores expressos em segundos

O protocolo TCP modificado a partir da técnica do par de pacotes permitiu que a aplicação, neste caso a uma simples transferência de arquivos, utiliza-se a informação de capacidade do enlace a tempo de melhorar sua performance. Com a informação sobre a capacidade do enlace, o protocolo TCP alocou o tamanho da janela de congestionamento de acordo com a capacidade do enlace, evitando assim a necessidade de reinício do tamanho da janela, possibilitando que os arquivos fossem transferidos de forma mais rápida em diferentes enlaces e localidades, provando que uma aplicação pode utilizar dessa informação para melhorar sua eficiência.

8 RESULTADOS DOS EXPERIMENTOS COM O USO DO GERADOR DE GRÁFICOS - PATHCRAWLER

Os experimentos descritos no capítulo 7 mostram como as aplicações poderiam utilizar da informação prévia sobre a capacidade dos enlaces para melhorar suas performances. Os experimentos descritos neste capítulo possuem o diferencial de mostrar de forma gráfica os resultados obtidos pela ferramenta que usa a técnica do par de pacotes para estimar a capacidade. Os resultados mostram de forma mais eficiente a capacidade de todos os enlaces envolvidos no caminho. Os testes foram feitos usando a rede de dados do Exército, aproveitando melhor a estrutura localizada na cidade do Rio de Janeiro, pois possui uma rede própria completa.

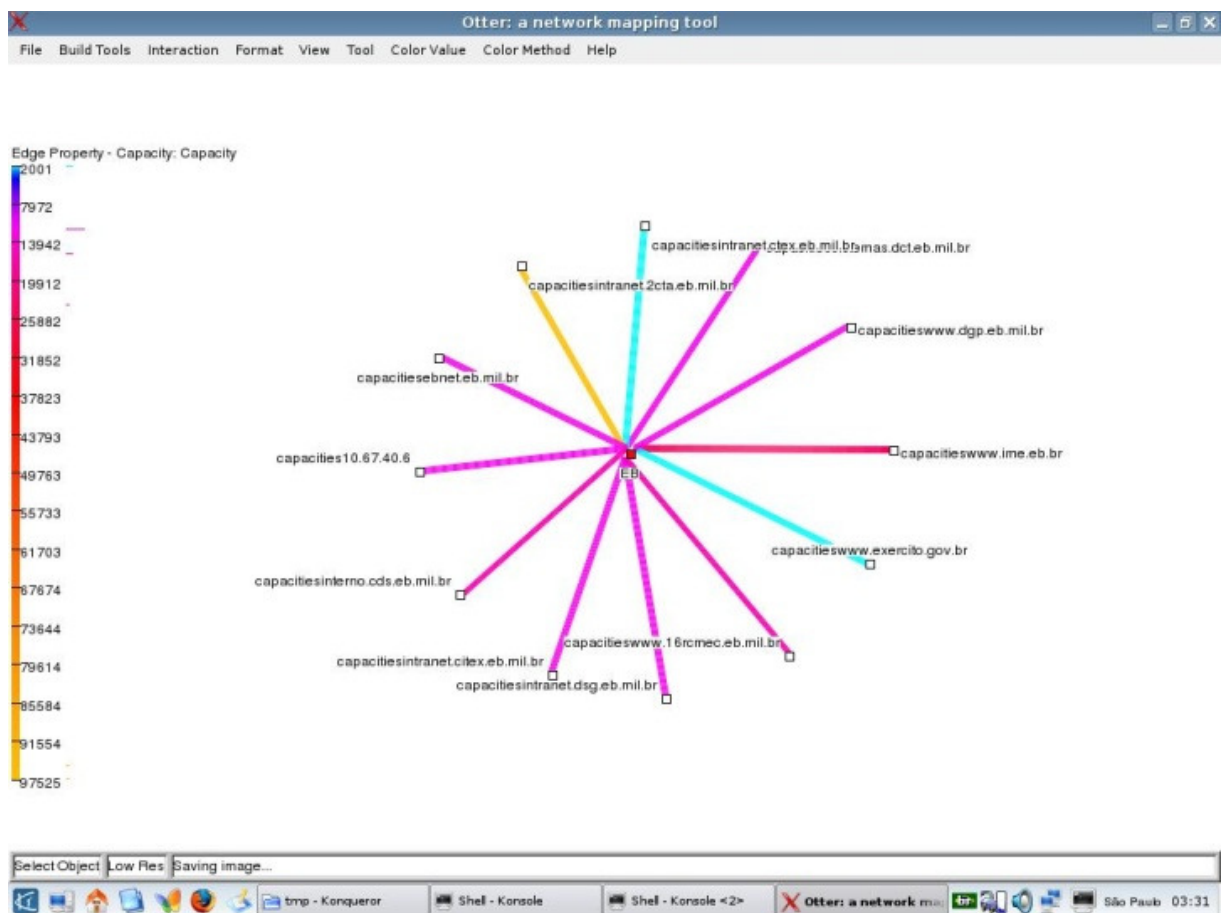


Figura 9 – Experimentos dentro da rede do Exército

Com o objetivo de fornecer maior número de dados para testes, foram feitos testes usando a Internet com enlaces de capacidade previamente conhecidos a fim de validar a informação resultante dos experimentos com os valores reais de capacidade do enlace. Essa informação é crucial para a administração de redes com elevadas ramificações o que ocorre muito com provedores de acesso.

Na realização do teste resultado na figura 9 foram escolhidos apenas pontos localizados dentro da rede do Exército, espalhados pelos estados do Rio de Janeiro e Brasília, ressaltando-se que da origem (EB) até o destino encontram-se vários enlaces com capacidades diferentes.

Para criar o visual gráfico foi usado o resultado obtido pelo método do par de pacotes a partir da ferramenta TCP-ECAP. Após a obtenção dos resultados, usa-se a ferramenta XPLOT para criar os gráficos. As diferentes capacidades são mostradas no gráfico no formato de cores, e os resultados foram validados com a capacidade real da rede que já era conhecida previamente.

Como exemplo tem-se o enlace de cor azul claro, com destino ao site www.exercito.gov.br, em que foi estimado um enlace de aproximadamente 2Mbps. Esta informação é verdadeira pois o gargalo desse caminho é um canal E1 com 2Mbps. As linhas de cor rosa representam uma saída do Rio de Janeiro para Brasília em um enlace MPLS contratado com capacidade de 16Mbps, que é o gargalo desse caminho. A linha próxima a cor vermelha, com destino ao www.ime.eb.br, é um enlace rádio que no momento dos testes estava funcionando com 21Mbps. E a linha próxima da cor laranja é uma máquina localizada na própria rede local com conexão fastethernet, por isso a capacidade próxima de 100Mbps.

A figura 10 apresenta experimentos realizados a partir de um enlace de saída com gargalo de 1Mbps diretamente ligado a Internet para diversos pontos com capacidades distintas.

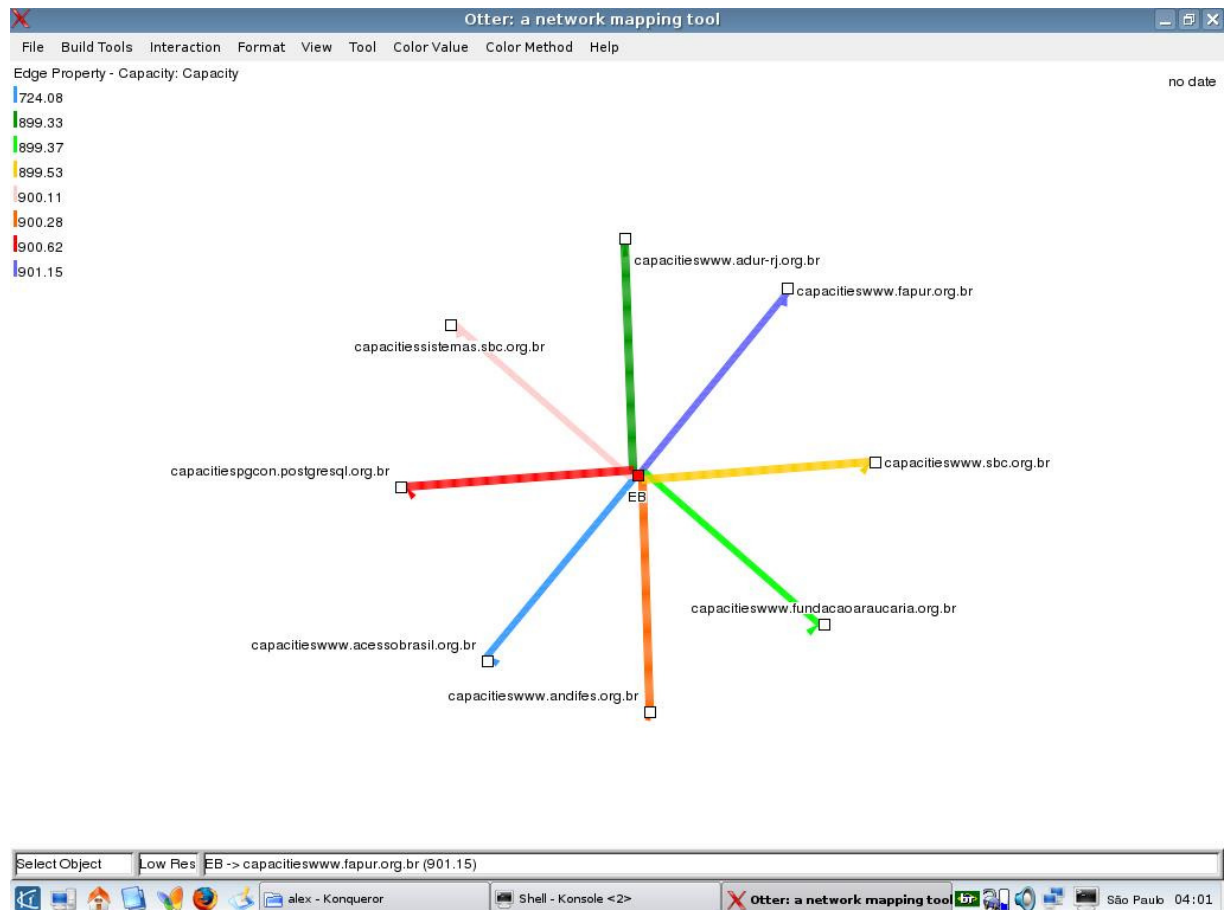


Figura 10 – Experimentos à Internet com enlaces de menor capacidade

Uma outra característica do *PathCrawler* é a obtenção do MinRTT, ou seja o menor tempo de ida e volta de um pacote a partir de mesmo ponto de origem. A figura 11 mostra estes tempos para oito diferentes localidades na casa dos milissegundos.

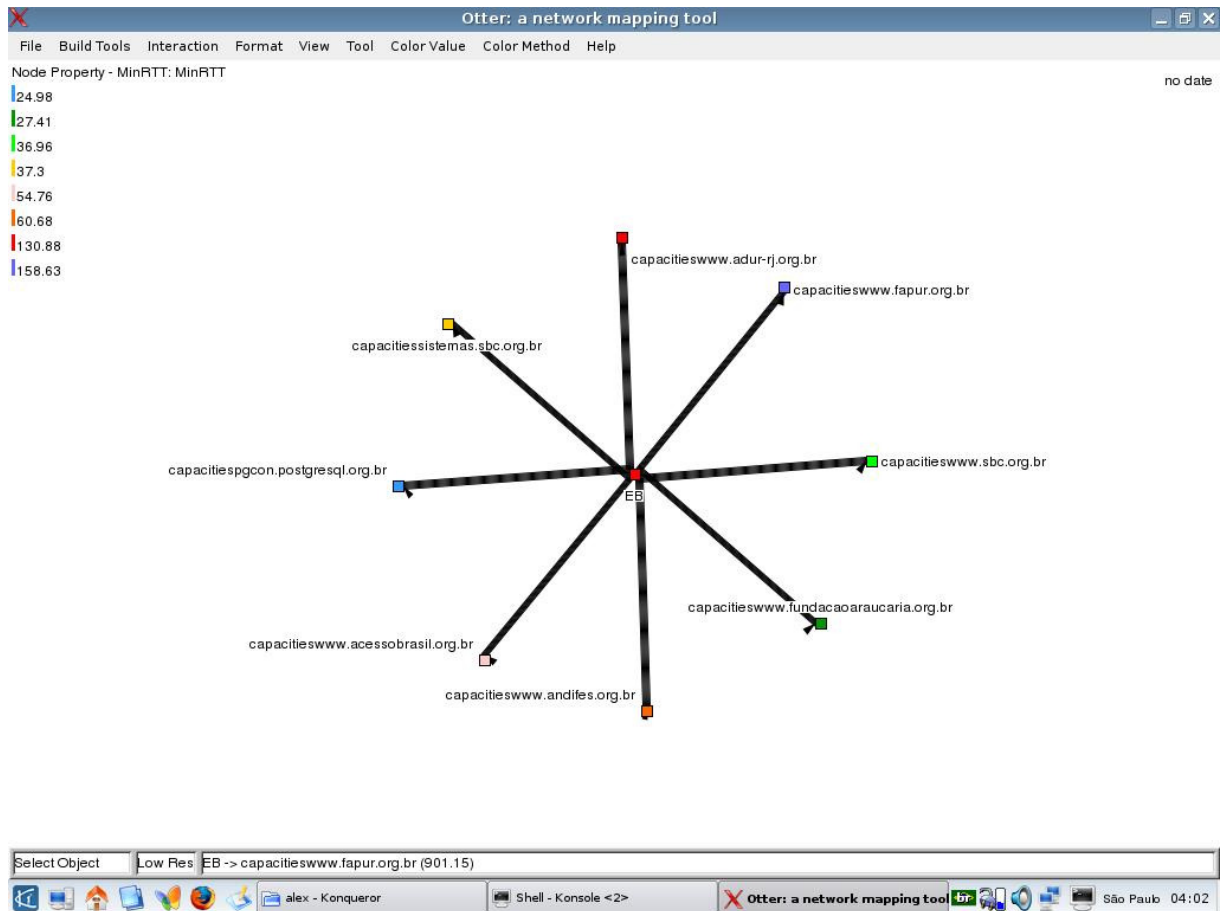


Figura 11 – Estimativa de MinRTT entre localidades

Outra forma de visualização dos resultados de estimativa de capacidade obtidos é pelo uso da ferramenta TCPTRACE, neste caso, na figura 12 obtemos de forma gráfica a informação que 95% dos enlaces explorados possuem capacidades menores que 2Mbps e que apenas dois enlaces possuem capacidade próxima a 100Mbps, ou seja uma fastethernet.

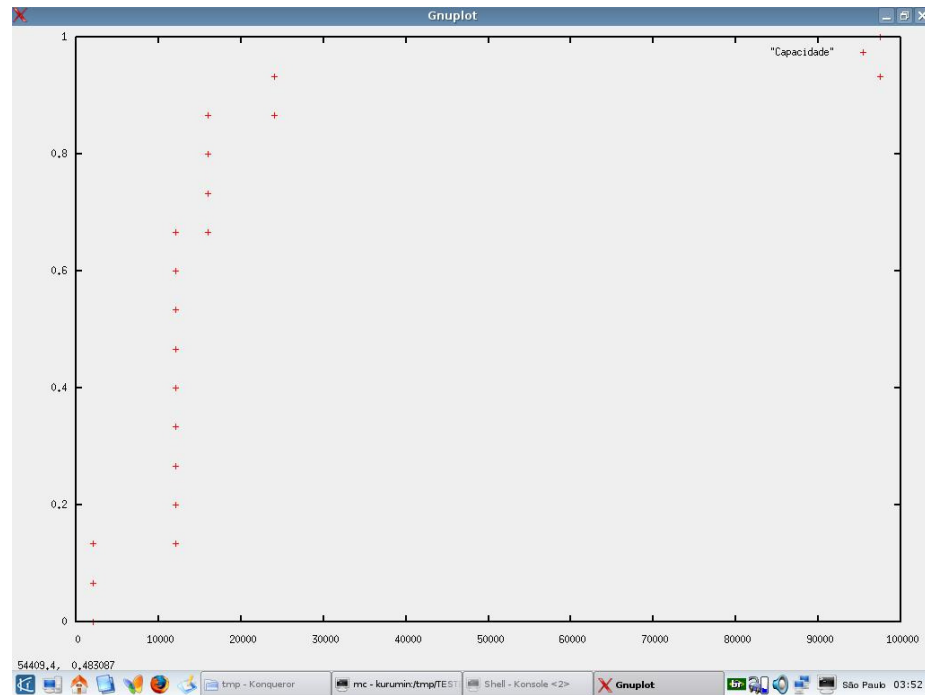


Figura 12 – Gráfico de distribuição das capacidades

A figura 13 apresenta o MinRTT de alguns enlaces e dentre os verificados, pouco mais de 60% possuem um valor menor que 100ms para o MinRTT.

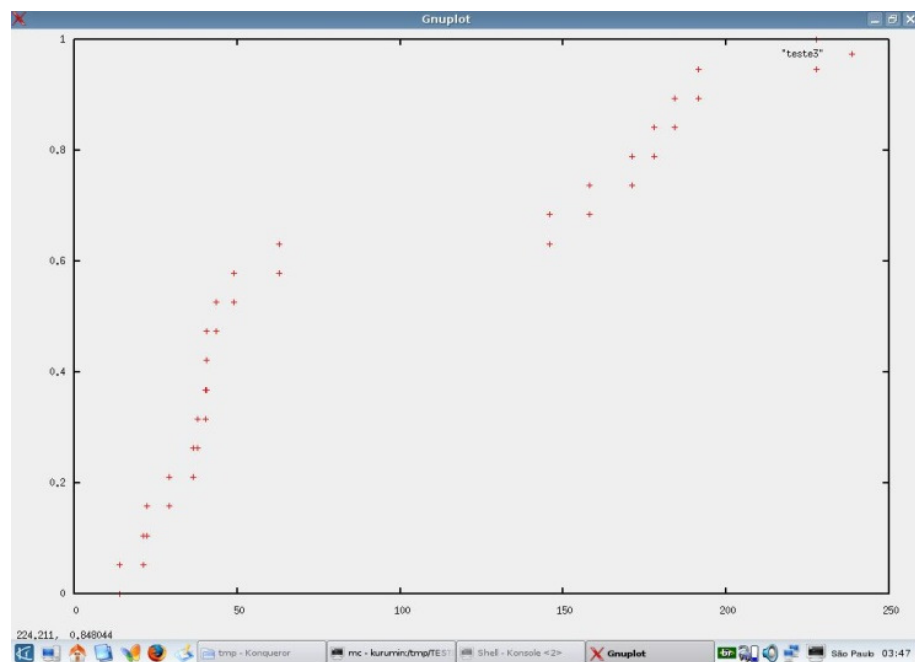


Figura 13 – Gráfico de distribuição MinRTT

9 CONCLUSÃO

Os testes foram conduzidos em horários de utilização normal da rede de dados, com o objetivo de melhor validar a ferramenta. Mesmo em horários de maior pico o TCP-ECAP mostrou uma melhora na performance do TCP, confirmando assim as expectativas. O comportamento do TCP-ECAP utilizado durante o funcionamento normal da rede foi menor que as performances de testes realizadas durante a noite. De acordo com testes anteriores, os resultados permaneceram em torno de 10% a 40% para downloads de arquivos em relação ao comportamento normal do TCP. Nos testes realizados com o TCP-ECAP, durante o funcionamento normal da rede, tais valores estavam na faixa de 10% a 20% em média, valores significativos em relação às implementações normais do TCP.

A portabilidade oferecida pelo ISO com Kernel 2.6.18.3 modificado gerou resultados significativos, pois ofereceu a instalação rápida da ferramenta e a possibilidade da utilização da ferramenta em diferentes localidades.

O método da estimativa de capacidade por par de pacotes ofereceu, em um tempo relativamente rápido, a informação necessária para que uma aplicação pudesse usufruir desses dados. Assim, foi melhorada a performance do TCP determinando o melhor tamanho para a janela de congestionamento.

O uso do *Pathcrawler* ofereceu uma nova perspectiva para os resultados obtidos pelo método de estimativa por par de pacotes. A forma gráfica como é apresentado o resultado permite uma utilização mais eficiente pelos administradores de redes. Uma melhoria a ser feita nesse caso deverá ser o tempo para montagem do gráfico, pois os resultados demoram um pouco para serem mostrados e essa característica pode ser um limitador, dependendo do objetivo para o qual se destina.

Os experimentos realizados em uma rede controlada foram essenciais para a validação do método e os resultados apresentados de acordo com os valores reais e capacidade dos enlaces.

10 TRABALHOS FUTUROS

O método da obtenção da estimativa de capacidade de um caminho pelo uso da técnica do par de pacotes, é muito referenciado em trabalhos de pesquisas sobre metrologia de redes. Como proposta futura deve-se verificar o comportamento deste método em enlaces mais específicos como enlaces de alta velocidade, e em situações onde temos a presença de enlaces rádios. Muitos dos rádios utilizados hoje funcionam a partir da tecnologia Wi-Max. Já existem estudos sobre o comportamento da técnica de estimativa de capacidade por par de pacotes em redes IEEE 802.11, porém poucos trabalhos referenciam o funcionamento desta técnica e redes IEEE 802.16.

11 REFERÊNCIAS

- [Carter 1996] CARTER, R. L. Measuring bottleneck link speed in packet switched networks. *Performance Evaluation*, v. 27, p. 297–318, 1996.
- [Jacobson 1997] JACOBSON, V. *Pathchar*: A tool to infer characteristics of internet paths. 1997. 241-250 p. Disponível em: <ftp://ftp.ee.lbl.gov/pathchar/>.
- [Kapoor et al. 2004] KAPOOR, R. et al. *Capprobe*: a simple and accurate capacity estimation technique. *ACM SIGCOMM Computer Communication Review*, v. 34, n. 4, p. 67–78, 2004.
- [Lai e Baker 1999] LAI, D.; BAKER, M. Measuring bandwidth. In: *Infocom-99*. [S.l.: s.n.], 1999.
- [Lai e Baker 2000] LAI, K.; BAKER, M. Measuring link bandwidths using a deterministic model of packet delay. In: *SIGCOMM*. [S.l.: s.n.], 2000. p. 283–294.
- [Martinello et al. 2007] MARTINELLO, M. et al. Exploring embedded path capacity estimation in tcp receiver. *SBRC*, 2007.
- [Scott 1992] SCOTT, D. *Multivariate Density Estimation: Theory, Practice and Visualization*. [S.l.]: Addison Wesley, 1992.
- [Tanenbaum 1996] TANENBAUM, A. S. *Computer Networks*. [S.l.]: PTR PH, 1996.
- [Lai e Baker 1999] LAI, D.; BAKER, M. Measuring bandwidth. In: *Infocom-99*. [S.l.: s.n.], 1999.
- [CAIDA 2005] Claffy, K: *Comparison of Public End-to-End Bandwidth Estimation Tools on High-Speed Links*, PAM, 2006
- [PARIS TRACEROUTE 2009] Teixeira, Renata: *Failure Control in Multipath Route Trace*, INFOCOM 2009